

Improving Energy Efficiency of OFDM Using Adaptive-Precision Reconfigurable FFT

Hatam Abdoli · Hooman Nikmehr ·
Naser Movahedinia · Florent de Dinechin

Received: date / Accepted: date

Abstract Being an essential issue in digital systems, especially battery-powered devices, energy efficiency has been the subject of intensive research. In this research, a multi-precision FFT module with dynamic runtime reconfigurability is proposed to trade off accuracy with the energy efficiency of OFDM in an SDR-based architecture. To support variable size FFT, a reconfigurable memory-based architecture is investigated. It is revealed that the radix-4 FFT has the minimum computational complexity in this architecture. Regarding implementation constraints such as fixed-width memory, a noise model is exploited to statistically analyze the proposed architecture. The required FFT word-lengths for different criteria - namely BER, modulation scheme, FFT size, and SNR - are computed analytically and confirmed by simulations in AWGN and Rayleigh fading channels. At run-time, the most energy-efficient word-length is chosen and the FFT is reconfigured while the required application-specific BER is met. Evaluations show that the implementation area and the number of memory accesses are reduced. The results obtained

H. Abdoli · H. Nikmehr · N. Movahedinia
Department of Computer Architecture, Faculty of Computer Engineering, University of Isfahan, 81746-73441 Isfahan, Iran
Tel.: +98-918-8510470
Fax: +98-31-36699529
E-mail: abdoli@eng.ui.ac.ir

H. Nikmehr
Tel.: +98-31-37934555
E-mail: nikmehr@eng.ui.ac.ir

N. Movahedinia
Tel.: +98-31-37934103
E-mail: naserm@eng.ui.ac.ir

F. Dinechin
Laboratoire CITI / INSA-Lyon Batiment Claude Chappe 6 Av. des Arts 69621 Villeurbanne France
Tel.: +33-472437430
E-mail: Florent.de-Dinechin@insa-lyon.fr

from synthesizing basic operators of the proposed design on an FPGA show energy consumption experienced a saving of over 80%.

Keywords Quantization noise · Reconfigurable design · FFT · OFDM · BER · Energy efficiency

1 Introduction

Over the past decades, wireless communication devices, such as mobile phones have rapidly developed from voice-only into smart-phones. Such an evolution is not possible unless requirements including higher computational performance, higher flexibility in programmability (to reduce production costs), greater efficiency in energy consumption and higher design reusability across some applications are appropriately met [50].

To use the radio-frequency spectrum more efficiently, new communication techniques, such as orthogonal frequency division multiplexing (OFDM) have been proposed, each with its own rapidly increasing standards and modes [46]. To be supported by communication devices, cost-effective and practical approaches to reuse the same hardware resources for different standards/modes need to be developed. To realize this reusability, the system hardware should be partly/entirely flexible (reconfigurable) and/or reprogrammable. The required flexibility, which significantly contributes to cost reduction, introduces software defined radio (SDR) solutions in wireless technology.

The functionality of a portable radio device not only depends on its battery capacity but also on the device's energy consumption behavior. The gap between the battery capacity delivered to and the power demands from such devices is expected to widen since the former is estimated to improve by about 10% per year, while the computational performance needs to increase ten times every five years [47].

Extensive use of energy-hungry applications such as video games, mobile P2P, video sharing, mobile TV, and 3D services signifies the energy consumption crisis. Therefore, research in this prominent issue [15,14] that considers the trade-off between design flexibility and energy efficiency is increasingly needed. Hence, to find the best scheme for the trade-off, different approaches such as designing architectures well-matched to algorithms and their run-time requirements may decrease the energy gap in SDR architectures.

OFDM has increasingly been adopted not only in the SDR architectures of wireless communications but also in wired and optical transmitters since it efficiently handles the frequency selective fading phenomenon, is robust against narrowband interference and impulse noise and provides an enhanced channel capacity. Nowadays, many digital communication standards and systems including UWB, WiMAX, WLAN, LTE, DVB-T, and ADSL have agreed to adopt OFDM standard [20].

As shown in Table 1, in the heart of most OFDM-based systems, the ubiquitous fast Fourier transform (FFT), with various sizes (N) and throughputs, is required. Designing an energy-efficient reconfigurable FFT system suitable

Table 1 System parameters for OFDM applications [44]

Standard	FFT size (N)	Sampling rate (MHz)	Bandwidth (MHz)
WLAN (802.11 a, n)	64, 128	20	20
WiMax (802.16e)	128, 512, 1024, 2048	1.43, 5.72, 11.43, 22.86	1.25, 5, 10, 20
3GPP-LTE	128, 256, 512, 1024, 1536, 2048	1.92, 3.86, 7.68, 15.36, 23.04, 30.72	1.25, 2.5, 5, 10, 15, 20
DAB	256, 512, 1024, 2048	2	1.5
DVB-T	2048, 8192	8	6, 7, 8

for different applications and standards has become an intriguing topic of study [19,1,37]. Optimum implementation of the FFT block in DSP systems may seriously improve their essential design aspects, such as energy efficiency, speed, and accuracy.

The next generation of wireless communication systems (5G) and cognitive radio (CR) devices [21] intend to significantly and dynamically improve spectrum usage, data rates, and energy efficiency. NC-OFDM (Non-Contiguous OFDM) [41] is a promising spectral agile variant of OFDM for dynamic spectrum aggregation [4]. According to the fast-changing wireless systems where various standards/modes should be supported, NC-OFDM CR systems need to change modulation schemes, the number of subcarriers, and FFT/IFFT size at run-time [16].

Traditionally, wireless transceivers are designed to deal with the worst-case operating conditions. Particularly in baseband digital processing units, data and operator word-lengths are chosen to be large enough (oversized) to cope with the worst unfavorable system conditions, such as very low signal-to-noise ratio (SNR). However, by using adaptive word-length processing, the system word-length is changed dynamically at run-time in order to deal with system conditions and performance requirements [33]. The use of adaptive precision (word-length) compromises the system precision to improve energy efficiency when the error-tolerable system can work properly with less accuracy.

The objective of this study is to introduce a flexible design using a set of criteria, including FFT size (N), the required SNR, maximum tolerable bit error rate (BER), the proper modulation scheme, and energy considerations, in order to select a specific configuration from a set of pre-defined configurations for the system to operate accordingly. Optimized allocation of hardware resources to support different configurations and to select the best system parameters based on the selection criteria leads to significant reduction in the area and the energy consumption of the implemented design.

The flexibility can be achieved through resizing the word-length of the reconfigurable FFT operands. Reducing the word-length decreases the VLSI area, the energy consumption, and the FFT precision; however, the system needs to be able to tolerate some proportional errors [28]. Further, it is clear that reducing the precision and VLSI area of FFT leads to increase the speed (throughput) of processing. Based on the required accuracy, this research calculates the tolerable error for different conditions and determines the corresponding acceptable FFT word-length. Accordingly, there is a tradeoff between accuracy (error probability) and energy efficiency. The system can be

optimized by choosing appropriate parameters and configuration, based on the channel conditions and system properties.

The contributions of this article can be pointed out as

- Unlike the previous studies, our study takes into account the role of trivial multipliers in radix-8 and radix-16 when computing the FFT hardware complexity. Hence, the radix-4 representation with the lowest complexity is chosen.
- A reconfigurable memory-based FFT is proposed to provide the required flexibility for SDR.
- A noise analysis for the proposed architecture is presented, considering the implementation constraints. BER is expressed analytically based on SNR and signal to quantization noise ratio (SQNR), to find the minimum required FFT word-length in AWGN and Rayleigh fading channels.
- An energy-efficient memory organization is proposed to dynamically provide the required word-length of FFT.

The rest of this article is organized as follows. A brief review of the existing literature is presented in Sect. 2. The FFT algorithm and the proposed reconfigurable architecture are discussed in Sect. 3. The noise model, the statistical analysis of the variable length FFT, and the analytical expression of BER are explained in Sect. 4. In Sect. 5, the proposed multi-precision FFT implanted inside an OFDM receiver is simulated and the results are compared with the analytical results. In addition, energy saving is estimated in this section for different configurations. Sect. 6 concludes the article.

2 Literature Review

In DSP processors, word-length optimization is to find the best format of data processing in order to decrease the area and energy consumption of the digital system. Traditionally, most previous works have focused on fixed (finite) word-length optimization where the data format and operator word-lengths are chosen to deal with the worst case conditions, like a low SNR in a noisy channel in wireless communication systems. Word-length optimization problem can be divided into two sub problems: range analysis and precision analysis [31].

There are proposed three methods to compute the required range/precision of digital arithmetic circuits:

- Dynamic analysis, also known as simulation-based method, evaluates the data-flow graph (DFG) of the system; hence, it is not fast, scalable, and robust enough [32, 26, 12].
- Static analysis, also called analytical method, propagates inputs and errors through the DFG of the design and evaluates the outputs through statistical analysis [29, 7, 3].
- Hybrid schemes [42, 5], which take the advantages of both dynamic and static methods.

The authors in [54,27] propose a dynamic word-length optimization technique to improve energy efficiency. In their approach, the higher the SNR, the more number of the lower significant bits of the data paths is set to zero. However, this method requires dedicated training symbols and a high number of iterative operations to find the correct word-length. This increases the processing overhead and the number of iterative operations that consequently decrease energy efficiency.

In [34], the trade-off between energy and accuracy is investigated in an SDR platform and the FFT word-length is optimized in an OFDM system using intensive simulations. However, this scheme is limited to only 8- and 16-bit word-length processing and also it is not clear how the circuit switches between the two configurations at run-time. Furthermore, the adaptation procedure is applicable to only very few modulation schemes and coding rates. In this approach, the FFT hardware architecture is not explored and there is not enough detail on reconfiguration time overhead.

Recently, a dynamic word-length tunable FFT architecture for a wireless system is proposed where the word-length is scaled according to the run-time conditions such as SNR. In these scenario-oriented methodologies, data format refinement is satisfied based on the run-time conditions and the best working scenario for the current conditions is found, chosen and applied at run-time [29, 12].

An analytic dynamic precision scaling (DPS) method is adopted in [29] in order to find the best word-length for each stage of the pipelined FFT, which seeks to reduce the time needed to find the optimum word-length and to decrease the power consumption. However, this approach only applies to fractional bits, (the integer bits are not considered) and the word-lengths of the twiddle factors and the inputs are assumed to be the same. In addition, this architecture only supports fixed size FFT architectures (only for a 256-point FFT). Furthermore, the power consumed by SRAMs and the memory accesses are not considered in the analysis, either.

A simulation-based DPS OFDM receiver is proposed in [12]. This design improves energy efficiency by tuning the run-time processing word-length, based on periodic estimation of channel conditions while the required BER is satisfied. Nevertheless, to estimate the energy saving, only a cost function for the arithmetic operations is used and other important aspects of the design such as the memories and buses, the control unit and the reconfiguration overhead are not included in the evaluation.

In [8], a reconfigurable pipelined architecture is proposed to compute variable-size FFT. Since the authors identify that the major energy dissipation is consumed by memory, they reduce memory accesses by decreasing the number of FFT stages. However the data-path width is not multi-precision and is only limited to 24 bits with FFT size restricted to powers of two.

Lee et al. [30] present a low-area dynamic reconfigurable pipelined FFT architecture for wireless networks. The method employs a four-path multipath delay commutator (MDC) in order to increase FFT processing throughput and

power efficiency. The architecture only supports 64-, 128- and 256-point FFT while the reconfiguration time and circuit overhead are large.

A run-time reconfigurable FFT processor is presented in [17] to support various FFT sizes and throughputs of 3G and 4G wireless standards using a mixed-radix pipelined architecture. The processor is suitable for cognitive radio applications and improves energy efficiency by enhancing resource usage efficiency and power saving. However, the reconfiguration time overhead is relatively high and the arithmetic operations and data-path width are limited to 16 bits.

Overall, to the best of our knowledge, the required word-length (precision) of reconfigurable FFT in different OFDM applications with various parameters (N, BER, SNR, modulation scheme, channel type) are not investigated analytically and are not specified precisely in the simulation-based methods. Hence, in this paper, these word-lengths are scrutinized analytically and verified by simulation. A feasible hardware architecture is also developed to implement the proposed multi-precision reconfigurable memory-based FFT with comparably low reconfiguration time overhead.

3 Proposed Memory-Based FFT Architecture

The pervasive FFT block is one of the main modules in DSP Processors, and more specifically in communication devices. The FFT is a fast implementation for discrete Fourier transform (DFT) expressed as [35]

$$x_k = \sum_{n=0}^{N-1} d_n \exp\left(-\frac{j2\pi nk}{N}\right) \quad (k = 0, 1, 2, \dots, N-1) \quad (1)$$

where d_n represents the input signal samples in the time domain, x_k is their output in the frequency domain and N is the number of FFT points (FFT size).

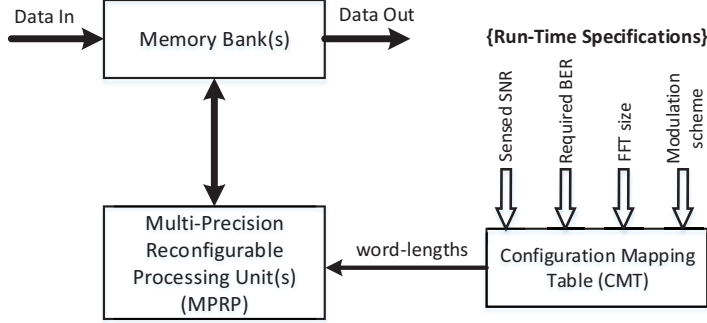
3.1 Overall Structure

Although there are very few approaches focusing on the floating-point FFT [2, 38], most of the energy-efficient designs concentrate on the fixed-point FFT, mostly due to its less complexity and lower power dissipation. In the multi-mode systems, it seems to be more effective to have a reconfigurable FFT unit in order to achieve the required word-length, accuracy, speed and consequently, more efficient energy consumption.

Common major FFT architectures used in OFDM systems are compared in Table 2. The comparison includes reconfigurability (flexibility) and hardware complexity of the FFT since these parameters greatly affect energy efficiency of the SDR system. The table shows that among the three FFT design approaches, the memory-based architecture [6] can be the best choice for a reconfigurable word-length FFT. In the other two, the fully-parallel architecture

Table 2 Comparison of well-known architectures in implementing the FFT

Architecture	Area	Throughput	Control complexity	Reconfigurability
Memory-based	Very low	Moderate	High	High
Fully-parallel	Very high	Very high	Low	NA (fixed)
Pipeline	High	High	Moderate	Very low

**Fig. 1** Proposed RMBFFT architecture

does not provide any flexibility at all and the pipeline architecture supports only a low degree of reconfigurability on the FFT size.

Although some designs are developed for reconfigurable pipeline FFT processors [44,51,48,10,24], their flexibilities are limited only to very few FFT sizes and as a result, the hardware overhead incurred by the reconfigurability is not tolerable for low-power schemes.

As shown in Fig. 1, the main components constructing the proposed reconfigurable memory-based FFT (RMBFFT) are memory bank(s), a configuration mapping table (CMT), and a multi-precision reconfigurable processing unit (MPRP). In different situations, MPRP can be reconfigured for smaller area to consume less energy while providing the required precision. In high throughput systems, the proposed RMBFFT employs more MPRP instances to increase processing speed. In Fig. 1, the memory banks are used to store RMBFFT inputs and outputs in every processing stage, which will be explained in more details in Sect. 5. In the proposed scheme, the best energy-efficient configuration (corresponding to the word-lengths) is obtained from CMT which stores the run-time specifications. These specifications such as the FFT size (N), the modulation scheme, sensed SNR, and required BER are applied while the system is working. The sensed SNR value is supposed to be measured using an SNR estimator block.

Using RMBFFT architecture makes the design of the FFT with any size of power of two ($N = 2^n$) possible. In the proposed RMBFFT architecture, intrinsically the memory width (word-length) is fixed in all stages. Consequently,

the number of output bits of the processing unit has to be the same as the number of input bits. This reduces the hardware cost as well as the energy consumed by the control unit.

As shown in Fig. 1, CMT maps the run-time specifications of the FFT to a word-length for RMBFFT. The entries of CMT are computed based on the required SQNR in order to guarantee the demanded BER. This means that increasing SNR and decreasing the word-length may lead to a configuration that makes RMBFFT less accurate but more energy-efficient. The entries of CMT are calculated analytically and verified by simulation, as described in Sect. 4 and Sect. 5.

In the proposed RMBFFT architecture, MPRP consists of the butterfly unit and the corresponding twiddle factor multipliers. To make MPRP energy-efficient through reconfigurability, modular multipliers and adders can be constructed using basic/small multipliers and adders. This provides the required flexibility to support all the operations on operands with different word-lengths. Like that in any arithmetic unit, in the butterfly unit the computation radix determines how many input bits are used to calculate the same number of output bits. Increasing the radix (usually a power of two) can potentially increase the processing speed and decrease the number of memory accesses in the expense of larger and more complex butterfly unit.

3.2 Radix-4 Butterfly

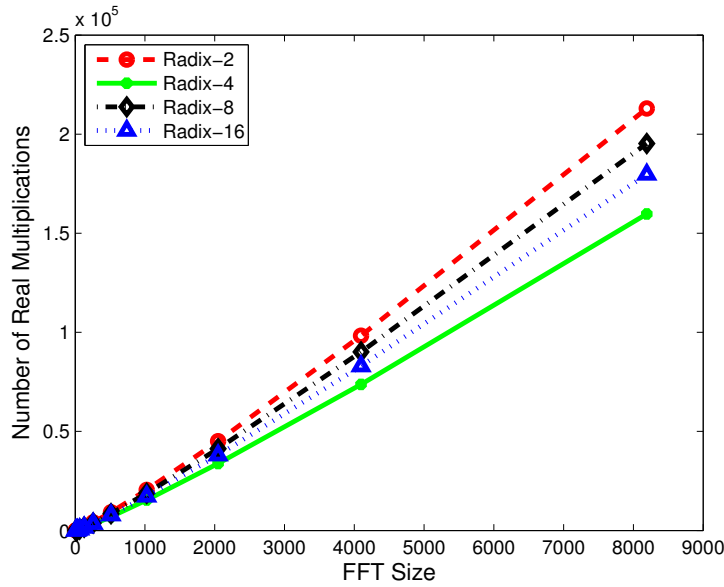
The common radices in the conventional FFT designs are 2, 4, and 8. When FFT size is not a power of 4 or 8, the mixed-radix-4/2 or 8/2 can be used to calculate all the required FFT size. For example with the FFT size of 32 in the mixed-radix-4/2, the FFT consists of two radix-4 stages followed by the last mixed-radix-4/2 reconfigures into radix-2 butterflies.

According to the previous research, to find the best computation radix that guarantees a low-power RMBFFT, the hardware complexities for different radices need to be examined using the entries of Table 3 where the numbers of non-trivial multiplications and additions are listed for different radices and various FFT sizes.

As indicated in the table, increasing the radix results in reduction of the overall number of multiplications and additions; hence the hardware complexity decreases that may increase energy efficiency. In previous studies, the trivial multiplications such as constant multiplications by $\frac{\sqrt{2}}{2}$ in the radix-8 butterfly are omitted when evaluating the hardware complexity. However, in reality, the complexity of such multipliers is about half the complexity of a non-trivial multiplier. This becomes even more important when N increases and the constant multipliers significantly affect the overall hardware complexity of the FFT architecture. Hence, the role of the trivial multiplications in the radices 8 and 16 is taken into account in the current research, unlike that in other studies. The comparison of the number of required multiplications for various radices of FFT, considering the trivial multiplications is illustrated in Fig. 2.

Table 3 Number of non-trivial real multiplications and additions for N-point FFT [39]

N	Number of real multiplications			Number of real additions		
	Radix-2	Radix-4	Radix-8	Radix-2	Radix-4	Radix-8
16	24	20	NA	152	148	NA
32	88	NA	NA	408	NA	NA
64	264	208	204	1032	976	972
128	712	NA	NA	2504	NA	NA
256	1800	1392	NA	5896	5488	NA
512	4360	NA	3204	13566	NA	12420
1024	10248	7856	NA	30728	28336	NA
2048	23560	NA	NA	68616	NA	NA

**Fig. 2** Total number of real multiplications (including trivial and non-trivial) vs. FFT size for different radices of FFT

As it is obvious in the figure, calculations in the radix-8 and above require more trivial multipliers to be used in the FFT design. Hence, using the radix-4 which results in the lowest hardware complexity for multiplication implementation can be the best choice for the proposed FFT architecture. Therefore, in Fig. 1, MPRP unit can be designed in the mixed-radix-4/2 that can be reconfigured into either the radix-4 or the radix-2 by the control unit based on N .

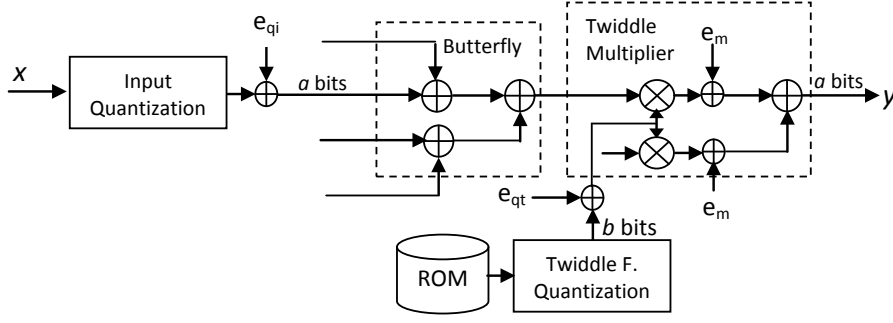


Fig. 3 Inserting quantization and rounding noise in DIF MPRP unit

4 Analytical Analysis of RMBFFT

In this section, SQNR of RMBFFT is formulated and then different values for BER are expressed analytically for different modulation schemes based on SNR and SQNR. Also, the tradeoff between SQNR and SNR is investigated. With dynamic adaptation of RMBFFT based on the sensed SNR, energy efficiency can be improved by reducing the word-length of RMBFFT, while the expected application-specific BER is guaranteed.

4.1 Quantization Loss and SQNR in RMBFFT

Based on the implementation constraints for the decimation in frequency (DIF) radix-4 RMBFFT introduced in Sect. 3, a noise model is developed for MPRP unit as shown in Fig. 3. In this figure, the word-lengths of both the memories and the arithmetic units are assumed to be the same. In the radix-4 butterfly unit, every real or imaginary output is calculated by summation of all four inputs. Hence, to prevent the possible overflow after this addition, the FFT inputs and the multiplication outputs in every stage are scaled down by shifting two bits to the right. This eliminates the noise caused by rounding in butterfly unit (rounding noise). Fig. 3 represents only the real part of the operations since the calculation for the imaginary part is the same.

In Fig. 3, x and y , respectively, represent the input and the output of one stage of the reconfigurable radix-4 FFT (MPRP), both with a -bit word-lengths. Also, e_{qi} and e_{qt} represent the additive quantization noise for the inputs of every stage and for the twiddle factors, respectively. In this scheme, the twiddle factors are stored in a ROM with b -bit word-length. For every output, the real phase of the complex multiplications consists of two real multiplications. Each of the real product terms is rounded to a bits and the additive rounding noise is represented by e_m . All the additive noises are assumed to be the uniformly distributed random variables with zero mean ($\mu = 0$) and non-correlated with one another; hence, their variances under round to nearest rounding scheme can be calculated as, $\sigma^2 = \frac{q^2}{12} = \frac{2^{-w}}{12}$, where w is the

word-length of the corresponding variable [39]. Every quantized input (\hat{x}) with word-length a , can be represented as $\hat{x} = x + e_{qi}$, where x is any ideal (unquantized) input and e_{qi} is the corresponding additive quantization noise [7]. In Fig. 3, without any noise (ideal situation), the output (y) can be expressed as, $y = 8x.t$, where t is the twiddle factor. The rounded output (\hat{y}) can be expressed using the quantized inputs and the additive rounding noise after each multiplication as:

$$\hat{y} = Q[y] = 4\hat{x}\hat{t} + e_m + 4\hat{x}\hat{t} + e_m \quad (2)$$

Consequently, the overall output quantization noise of one stage of the DIF FFT can be calculated as

$$n_{DIF} = \hat{y} - y = 8(xe_{qt} + te_{qi} + e_{qi}e_{qt}) + 2e_m \quad (3)$$

which leads to the quantization noise variance of

$$\sigma_{n_{DIF}}^2 = E\{|n_{DIF}|^2\} = \frac{64}{12}(2^{-2b}x^2 + 2^{-2a}t^2 + 2^{-2a-2b}) + \frac{4}{12}2^{-2a} \quad (4)$$

From Eq. 4 and assuming x and t to be non-correlated variables normalized in the range $[-1, 1]$ with uniform distribution, it can be concluded that $\sigma_x^2 = \sigma_t^2 = \frac{1}{3}$. Accordingly, Eq. 4 can be rewritten as

$$\sigma_{n_{DIF}}^2 = \frac{64}{12} \left(\frac{1}{3}(2^{-2b} + 2^{-2a}) + 2^{-2a-2b} \right) + \frac{1}{3}2^{-2a} \quad (5)$$

The scaling factor of $1/4$, caused by the 2-bit scaling noted in Sect. 4.1, reduces the variance of the quantization error by a factor of $(\frac{1}{16})^{n-i}$ in the i -th stage ($i = 1, 2, \dots, n$) where $n = \log_4 N$ is the number of FFT stages [39]. Consequently, the total variance of the output quantization noise of RMBFFT is

$$\begin{aligned} \sigma_q^2 &= \sigma_{n_{DIF}}^2 \left\{ \left(\frac{N}{4}\right)\left(\frac{1}{16}\right)^{n-1} + \left(\frac{N}{4^2}\right)\left(\frac{1}{16}\right)^{n-2} + \dots + \frac{N}{4^n} \right\} \\ &= \sigma_{n_{DIF}}^2 \left\{ \left(\frac{1}{4}\right)^{n-1} + \left(\frac{1}{4}\right)^{n-2} + \dots + 1 \right\} = \frac{4}{3}\sigma_{n_{DIF}}^2 \left\{ 1 - \left(\frac{1}{4}\right)^n \right\} \end{aligned} \quad (6)$$

For sufficiently large N , Eq. 6 can be approximated as $\sigma_q^2 = \frac{4}{3}\sigma_{n_{DIF}}^2$. With regard to the variance of RMBFFT output signal as $\sigma_y^2 = \frac{1}{3N}$ [39], due to the scaling in every stage, SQNR of RMBFFT can be expressed as

$$SQNR = \frac{\sigma_y^2}{\sigma_q^2} = \frac{3}{12N\sigma_{n_{DIF}}^2} \quad (7)$$

The above analysis is developed for the radix-4 DIF FFT. The quantization noise analysis for the decimation in time (DIT) FFT method is similar to what is given in Fig. 3, since DIT and DIF are similar except for the multipliers in

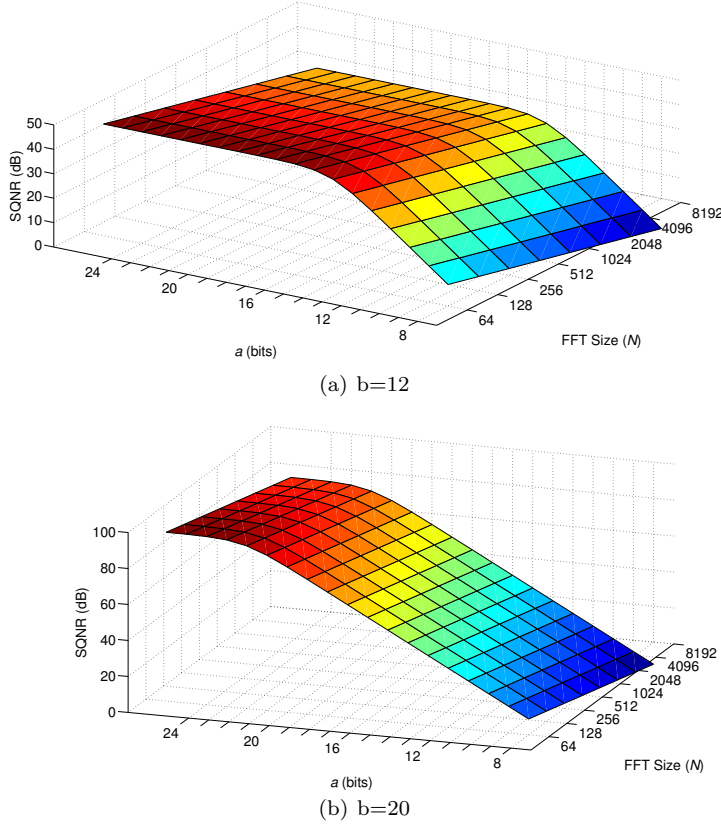


Fig. 4 SQNR vs. the FFT size (N) and input word-length

DIT that are moved to the input of the butterfly unit. Hence, the overall noise variance for the radix-4 DIT FFT can be expressed as

$$\sigma_{n_{DIT}}^2 = \frac{64}{12} \left(\frac{1}{3} (2^{-2b} + 2^{-2a}) + 2^{-2a-2b} \right) + \frac{64}{12} \left(\frac{1}{3} 2^{-2a} \right) \quad (8)$$

Based on the implementation constraints of the radix-4 RMBFFT, comparing Eqs. 5 and 8 indicates that the quantization noise variance of the DIF scheme is smaller than that of the DIT design. Fig. 4 shows how choosing different values for N , a and b can affect SQNR. The figure reveals that increasing N and/or decreasing a can increase SQNR. As Fig. 4 indicates, situations in where inputs word-lengths (a), are larger than the twiddle factor word-length (b), do not considerably increase SQNR. The same results could be obtained once a and b are interchanged in the above analysis. This means that for twiddle factors word-length larger than inputs word-lengths, SQNR does not increase substantially.

4.2 BER Analysis in AWGN Channel

The aim of this section is to determine the minimum required accuracy (word-length) of RMBFFT for different SNR values, modulation schemes, FFT size and application-specific BERs. The maximum reduction in SQNR to trade off accuracy for energy efficiency is calculated analytically, using the statistical analysis of RMBFFT. In this approach, at run-time, a special module is used to measure SNR and then to send this value to the FFT control unit. This unit then selects the word-lengths from CMT based on the SNR range that the sensed SNR falls into.

In this section $c(k)$ is the additive white Gaussian noise (AWGN) channel with the noise variance of $\sigma_c^2 = \frac{P_{signal}}{SNR}$. Its probability density function (pdf), $f_c(x)$ can be expressed as

$$f_c(x) = \frac{1}{\sigma_c \sqrt{2\pi}} \exp\left(\frac{-x^2}{2\sigma_c^2}\right). \quad (9)$$

The overall quantization noise to signal ratio, $\sigma_Q^2 = \frac{1}{SQNR}$ with SQNR calculated in Sect. 4.1, includes additive noises, e_{qi} , e_{qt} , and e_m , which are assumed to be independent and identically distributed (iid). Hence, the overall sum can be considered as a Gaussian variable with the pdf function in the form of

$$f_q(x) = \frac{1}{\sigma_Q \sqrt{2\pi}} \exp\left(\frac{-x^2}{2\sigma_Q^2}\right). \quad (10)$$

The pdf function can be represented as the convolution of the variables pdf functions. Hence, convolving Eqs. 9 and 10 results in

$$f_N(x) = \frac{1}{\sqrt{\sigma_Q^2 + \sigma_c^2} \sqrt{2\pi}} \exp\left(\frac{-x^2}{2(\sigma_Q^2 + \sigma_c^2)}\right). \quad (11)$$

The probability distribution $F_N(x)$ can be extracted from Eq. 11 as

$$F_N(x) = \frac{1}{2} \left(1 + \operatorname{erf}\left(\frac{x}{\sqrt{2(\sigma_Q^2 + \sigma_c^2)}}\right) \right) \quad (12)$$

where $\operatorname{erf}(x) = \frac{2}{\sqrt{\pi}} \int_0^x \exp(-t^2) dt$.

Each OFDM symbol contains overheads in both time domain (cyclic prefix) and frequency domain (guard bands) which take various values in different application layer standards. Although these overheads affect SQNR, they are ignored and only the effects of FFT module and modulation schemes are considered when calculating BER. In BPSK and QPSK modulations, BER can

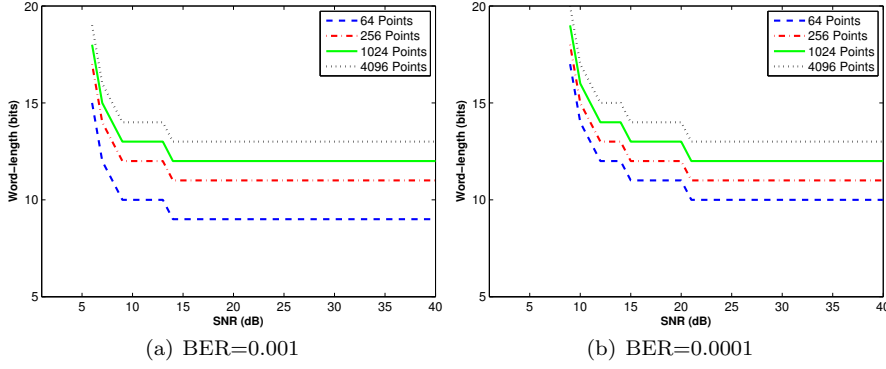


Fig. 5 Required input word-length vs. SNR for different FFT sizes in QPSK (AWGN)

be calculated through

$$BER(\sigma_Q, \sigma_c) = 1 - F_N(1) = \frac{1}{2} \operatorname{erfc} \left(\frac{1}{\sqrt{2(\sigma_Q^2 + \sigma_c^2)}} \right) \quad (13)$$

where $\operatorname{erfc}(x) = 1 - \operatorname{erf}(x) = \frac{2}{\sqrt{\pi}} \int_x^\infty \exp(-t^2) dt$.

BER for 16-QAM modulation can be calculated using [18]:

$$BER(\sigma_Q, \sigma_c) = \frac{3}{8} \operatorname{erfc} \left(\frac{1}{\sqrt{10(\sigma_Q^2 + \sigma_c^2)}} \right) + \frac{1}{4} \operatorname{erfc} \left(\frac{3}{\sqrt{10(\sigma_Q^2 + \sigma_c^2)}} \right) - \frac{1}{8} \operatorname{erfc} \left(\frac{\sqrt{5}}{\sqrt{2(\sigma_Q^2 + \sigma_c^2)}} \right) \quad (14)$$

Based on BER obtained analytically through Eqs. 13 and 14, the required input word-lengths are calculated for different SNR to guarantee the application-specific BER constraint. The input word-lengths calculated for QPSK and 16-QAM modulations in AWGN channel are shown in Figs. 5 and 6, respectively. For the sake of clarity, the word-lengths are shown for the proposed 64-, 256-, 1024-, and 4096-point RMBFFT. As expected, by increasing SNR and/or decreasing N , even smaller word-lengths can satisfy the required BER. Decreasing the word-length in RMBFFT leads to a smaller arithmetic circuit with less energy consumption and with the accuracy which still remains in the acceptable range.

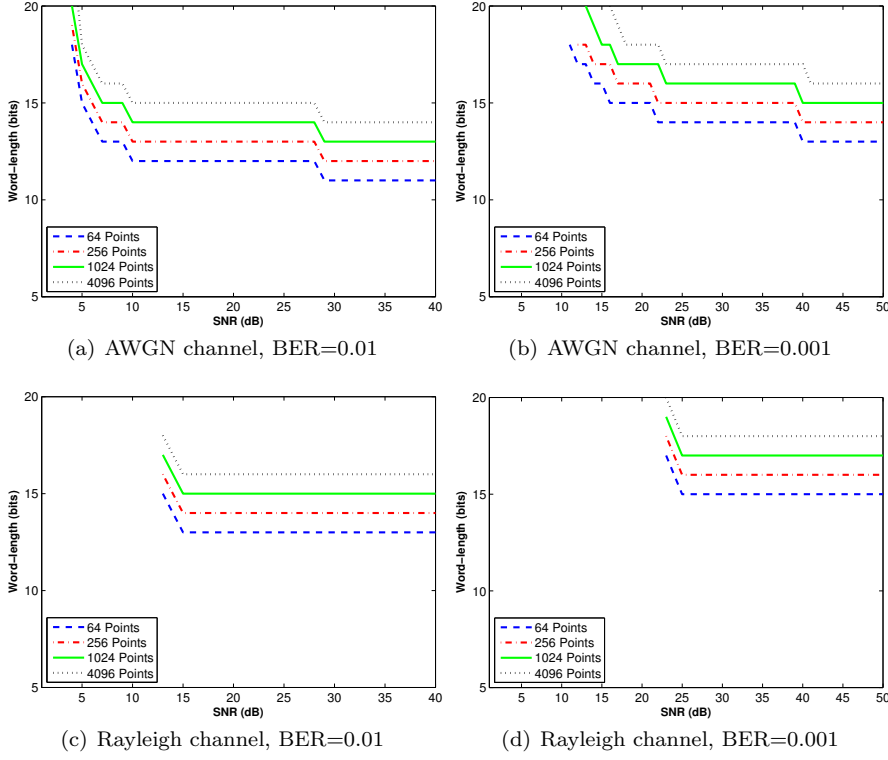


Fig. 6 Required input word-length vs. SNR for different FFT sizes in 16-QAM

4.3 BER Analysis in Rayleigh Fading Channel

This section presents BER expression to find the minimum required word-length of RMBFFT in a Rayleigh fading channel. The OFDM technique can transform a frequency-selective wide-band channel into many non-selective (flat) narrowband subchannels. To compute BER, the method in chapter 13 of [40] is exploited in Eqs. 15 and 16 to average the non-fading AWGN BER expression with the chi-squared distribution obtained in the frequency non-selective Rayleigh fading channel:

$$P_b = \int_0^\infty P_{b,AWGN}(\gamma) P_{df}(\gamma) d\gamma \quad (15)$$

$$p_{df}(\gamma) = \frac{1}{\bar{\gamma}} \exp\left(-\frac{\gamma}{\bar{\gamma}}\right) \quad \gamma \geq 0 \quad (16)$$

where $P_{b,AWGN}(\gamma)$ represents the probability of error (BER) of a specific modulation scheme in an AWGN channel at a particular SNR; $\gamma = h^2 \frac{E_b}{N_0}$. Here, $\frac{E_b}{N_0}$ is the ratio of bit energy to the noise power density in a non-fading AWGN channel, and the random variable h^2 is the instantaneous power of

the fading channel. In Eq. 16, $P_{df}(\gamma)$ is the pdf function of γ caused by the Rayleigh fading channel, and $\bar{\gamma} = \frac{E_b}{N_0} E[h^2]$ is the average SNR. If $E[h^2] = 1$, $\bar{\gamma}$ provides the average $\frac{E_b}{N_0}$ in the fading channel [45]. Using Eqs. 15 and 16 and also Eq. 17 [18],

$$\int_0^\infty a \cdot \exp(-at) \operatorname{erfc}(\sqrt{bt}) dt = 1 - \sqrt{\frac{b}{a+b}} \quad (17)$$

the average BER for M-QAM in a frequency non-selective Rayleigh fading channel can be expressed by [45]

$$BER_{M-QAM, Ray} \approx \frac{2}{\log_2 M} \left(1 - \frac{1}{\sqrt{M}}\right) \sum_{i=1}^{\frac{\sqrt{M}}{2}} \left(1 - \sqrt{\frac{1.5(2i-1)^2 \bar{\gamma} \log_2 M}{M-1+1.5(2i-1)^2 \bar{\gamma} \log_2 M}}\right) \quad (18)$$

Using both Eq. 18 and the AWGN BER of our RMBFFT (Eq. 14), the BER of 16-QAM modulation in a Rayleigh fading channel can be expressed as:

$$BER_{16-QAM, Ray} \approx \frac{1}{2} - \frac{3}{8} \frac{1}{\sqrt{1+10(\sigma_Q^2 + \sigma_c^2)}} - \frac{1}{4} \frac{3}{\sqrt{9+10(\sigma_Q^2 + \sigma_c^2)}} + \frac{1}{8} \frac{\sqrt{10}}{\sqrt{10+4(\sigma_Q^2 + \sigma_c^2)}} \quad (19)$$

where σ_Q^2 and σ_c^2 are defined in Sect.4.2.

Figs. 6(c) and 6(d) show the minimum required word-lengths calculated for 16-QAM modulation in the Rayleigh fading channel for BER=0.01 and BER=0.001, respectively. Similar to the AWGN, by increasing SNR and/or decreasing N , the minimum RMBFFT word-length can be reduced to satisfy the required BER in the Rayleigh model. Comparing Figs. 6(c) and 6(d) with Figs. 6(a) and 6(b) shows that the minimum FFT word-length requires more bits on the Rayleigh fading channel than on the AWGN channel when all the other parameters are the same.

5 Performance Evaluation

In this section, the analysis expressed in Sect. 4 is verified and compared with simulation results and the minimum required word-lengths are obtained for different specifications. Also, the reconfiguration overhead and the energy saving of RMBFFT are estimated. In accordance with RMBFFT, a memory organization is proposed to improve energy-efficiency.

5.1 Simulation Results

As shown in Fig. 7, the simplified OFDM transceiver is modeled in Simulink to validate the analysis performed in Sect. 4. In this figure, the generated random data are modulated in either QPSK or 16-QAM, the cyclic prefixes (CP), and subcarriers structure are based on LTE standard, and the channel model is AWGN or Rayleigh fading. At the receiver side, the proposed RMBFFT is

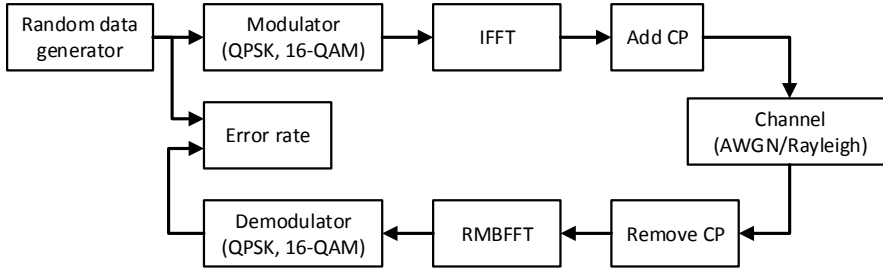


Fig. 7 Simplified OFDM transceiver model in Simulink

implemented in MATLAB for 8- to 20-bit word-lengths. The channel is swept from 5 to 40 dB and BER of the corresponding system is measured using the Error Rate module. For different SNR and FFT sizes, based on the demanded application-specific BER, the required word-length of RMBFFT is obtained through simulation and compared with the analytical results.

For the sake of clarity, these comparisons are shown only for 256- and 1024-point RMBFFT in Fig. 8 for both AWGN and Rayleigh fading channels. Because of the assumptions used in Sect. 4 in order to simplify the analysis, the analytical and simulation results are not completely the same. Nevertheless, the difference is not significant and the simulation results follow the same general trend. As indicated in Fig. 8, the required word-length rises when N (FFT size) increases and SNR decreases in both schemes.

From the simulations and analytical results, it can be concluded that for LTE, the required word-length for different specifications (N , BER, modulation schemes and SNR) can vary from 8 up to 19 bits. However, implementing a practical reconfigurable RMBFFT hardware with one bit step is not efficient. Instead, optimized 4-bit addition and multiplication blocks can be used to efficiently construct 8-, 12-, 16-, 20-bit RMBFFT iteratively using modular (recursive) technique in digital arithmetic [36].

At run-time, entries of CMT (shown in Fig. 1) that represent word-lengths of RMBFFT are selected by the system specifications which are SNR, BER, N , and modulation scheme. The entries labeled as “ wl ” (word-length) are listed in Table 4. The value of N in Table 4 is limited to 2048 points while in the other standards it may increase to 8192 or even more. For larger N and more complicated modulation schemes (like 64-QAM), larger RMBFFT word-lengths may be required.

5.2 Overhead Analysis

As mentioned in Sect. 4, at run-time a simple SNR estimator module that measures current SNR is required. Usually this module is already embedded and available in most wireless communication systems for different objectives such as: selecting modulation/coding parameters and channel state feedback [49].

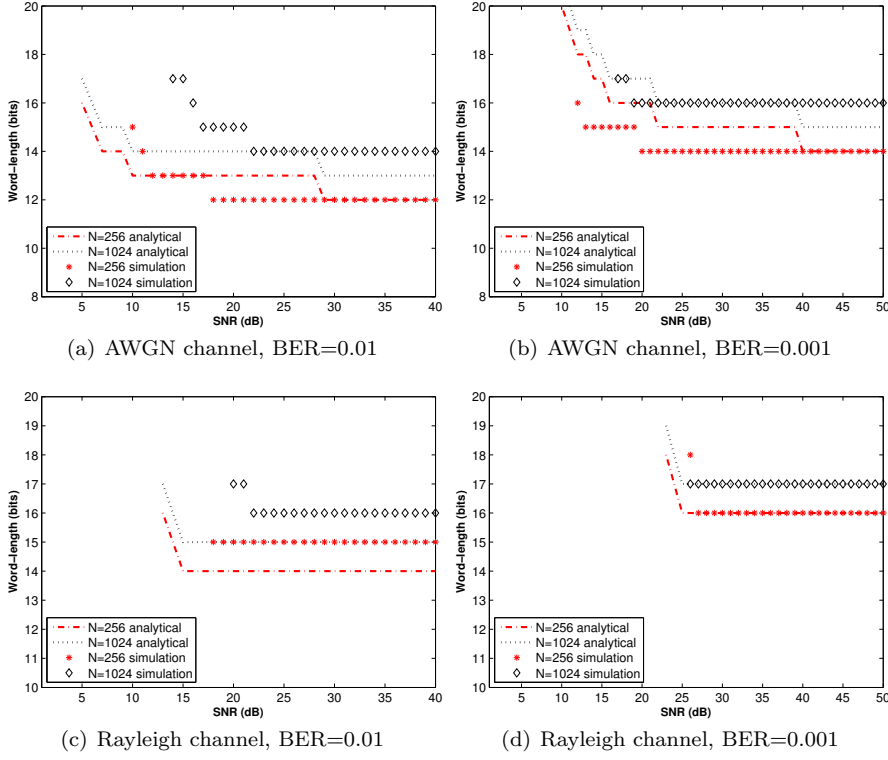


Fig. 8 Analytical-based vs. simulation-based required word-length for 16-QAM

Therefore, in the current analysis the overhead of SNR estimator block is not taken into account.

In the proposed reconfigurable design, CMT decision table provides the suggested word-length for MPRP unit of RMBFFT. To investigate the energy overhead of CMT in the proposed architecture, CMT and basic operators in MPRP are modeled in VHDL using FloPoCo [13] and synthesized on a Virtex-6 xc6vcx75t Xilinx FPGA. The power dissipation of CMT circuit (P_c), 4-bit adder (P_a), and 4-bit multiplier (P_m) in the modular MPRP unit are estimated using Xilinx Xpower Analyzer (XPA) [52]. Taking the fact into consideration that transition rate of input data and data access rate of CMT are considerably less than that of MPRP, the results are $P_c = 110 \mu W$, $P_a = 20 \mu W$, and $P_m = 90 \mu W$. The energy saving of the proposed RMBFFT for different word-lengths are estimated in the next section, using the obtained power dissipation.

5.3 Energy Saving Estimation

In modular (divide and conquer) design, addition (multiplication) of 8-, 12-, 16-, and 20-bit operands can be constructed by using 2 (4), 3 (9), 4 (16), and

Table 4 Content of CMT for LTE standard where “ wl ” represents word-length

BER=0.01												
QPSK							16-QAM					
N	SNR	wl	SNR	wl	SNR	wl	SNR	wl	SNR	wl	SNR	wl
64	<10	12	≥ 10	8			<8	16	≥ 8	12		
128	<8	16	≥ 8 <22	12	≥ 22	8	<17	16	≥ 17	12		
256	<8	16	≥ 8	12			≤ 10	20	> 10 <18	16	≥ 18	12
512	≤ 11	20	> 11	16			≤ 12	20	> 12	16		
1024	≤ 12	20	> 12	16			≤ 15	20	> 15	16		
1536	≤ 18	20	> 18	16			≤ 19	20	> 19	16		
2048	≤ 19	20	> 19	16			≤ 20	20	> 20	16		

BER=0.001												
QPSK							16-QAM					
N	SNR	wl	SNR	wl	SNR	wl	SNR	wl	SNR	wl	SNR	wl
64	≤ 6	16	> 6 <15	12	≥ 15	8	≤ 13	16	> 13	12		
128	≤ 15	16	> 15	12			> 13 <20	16	≥ 20	12		
256	<17	16	≥ 17	12			≤ 14	20	> 14 <20	16	≥ 20	12
512	≤ 13	20	> 13	16			≤ 18	20	> 18	16		
1024	≤ 15	20	> 15	16			≤ 20	20	> 20	16		
1536	≤ 22	20	> 22	16			≤ 23	20	> 23	16		
2048	≤ 24	20	> 24	16			≤ 25	20	> 25	16		

5 (25) iterations of 4-bit adder (multiplier) block, respectively. Referring to Fig. 3, each output of MPRP combinational logic consists of three additions in butterfly, and two multiplications and one addition in multiplier unit. In modular multiplication, once the multiplier and multiplicand are sliced into x 4-bit blocks, x^2 4-bit multipliers followed by x^2 8-bit adders are required to sum up the partial products. Similarly, a $4x$ -bit addition can be implemented by x instances of 4-bit adders.

Since the word-lengths of additions and multiplications are the same (a), the estimated energy consumption of butterfly and multiplier unit are $E_b = 3\frac{a}{4}P_a t_a$ and $E_m = \left(2\left(\frac{a}{4}\right)^2\right)P_m t_m + \left(2\left(\frac{a}{4}\right)^2 + \frac{a}{4}\right)P_a t_a$, respectively. The power dissipations P_a and P_m are explained in Sect. 5.2 and t_m and t_a are the critical path delays of 4-bit multiplier and 4-bit adder, respectively.

Since CMT is accessed once every execution of RMBFFT, energy consumption of CMT can be estimated as $E_c = P_c t_c$, where t_c is CMT critical path delay. So, based on the dynamic word-length extracted from CMT (a), the overall energy consumption of RMBFFT can be estimated as

$$E_R = E_m + E_b + E_c = \frac{a^2}{8}P_m t_m + \left(a + \frac{a^2}{8}\right)P_a t_a + P_c t_c \quad (20)$$

As shown in Fig. 9, to improve energy-efficiency of RMBFFT memory unit, the proposed interleaved memory is arranged as a 54 matrix of the memory banks. The memory organization consists of $(N/4)$ 4-bit modules to provide the ability to read/write nibbles of four inputs/outputs (A, B, C , and D in

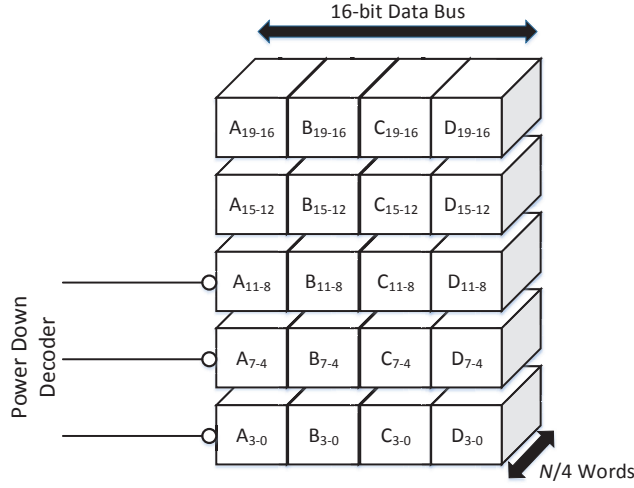


Fig. 9 Proposed energy-efficient memory organization of RMBFFT

the figure) for one radix-4 butterfly through a 16-bit data bus, iteratively. Based on the selected RMBFFT word-length, the memory banks are accessed consecutively, from MSB to LSB rows.

Since memory accesses are responsible for major part of energy consumed in FFT [43], reducing the number of memory accesses can improve the overall energy-efficiency considerably. In addition, when the required dynamic word-length decreases, energy efficiency of memory can be improved by forcing unused memory modules into idle state. For example, for 12-bit word-length, only the upper three rows of the banks are used and are accessed during three consecutive iterations. Hence, energy consumption can be reduced about 40%, comparing to 20-bit word-length case. Also, the banks in the two least significant rows can be powered down, which results in further 40% improvement in energy efficiency.

After being normalized to the conventional 20-bit FFT implementation, energy consumptions of different configurations of RMBFFT are compared and listed in Table 5. The maximum energy saving of MPRP is about 81% for the 8-bit word-length, while energy consumption rises to 103% for 20-bit due to the reconfiguration overhead penalty. The proposed memory organization having been used, up to 60% more energy efficiency is achievable due to reducing the number of memory accesses. Also, by disabling unused memory modules for smaller word-lengths, the energy consumption of modules can be reduced by 40% for the 8-bit RMBFFT.

Table 5 Normalized energy consumptions for different RMBFFT configurations

Energy Consumption of [%]	(Worst Case)	RMBFFT			
	Static 20-b	20-b	16-b	12-b	8-b
MPRP	100%	103%	67%	39%	20%
memory accesses	NA	100%	80%	60%	40%
memory modules (by disabling modules)	NA	100%	80%	60%	40%

Table 6 Performance comparison

Design	[12]	[30]	[17]	[24]	Proposed
FPGA device	Virtex-5	Virtex-5	Zynq-7000	Virtex-5	Virtex-6
FFT size	512	64, 128, 256, 512	64-2048	512	(2 ⁿ)
Architecture	NR	Pipeline MDC	Pipeline SDF	Pipeline	Memory-based
Radix	NR	Mixed-radix 4/2	Mixed-radix 2 ² /2/3	Radix 2 ⁵	Mixed-radix 4/2
Word length	4-15	16	16	20	{8,12,16,20}
Reconfigurability	No	Yes	Yes	Yes	Yes
Reconfiguration time (μ s)	NR	689	496	NR	0.01
Multi-precision	Yes	No	No	No	Yes
Core power (mW)	NR	NR	26-82	126	2P
Energy consumption per symbol (pJ)	40-110	NR	NR	NR	{7,15,26,40}
Maximum clock rate (MHz)	NR	111	NR	340	147
Throughput (MSample/s)	NR	400	98	8R	$P.R.(\frac{1}{wl})^2$
Latency - critical path (ns)	NR	8.959	NR	2.941	6.8
Number of slice registers	NR		6400	21427 (31%)	44P (1%)
Number of slice LUTs	NR	20670 (Overall)	3200	16589 (24%)	46P (1%)
Number of flip flops	NR		20 DSP core	26957 (39%)	46P (1%)

NR: Not Reported

R: clock rate

P: Number of MPRP units

5.4 Performance Comparison with Previous Works

Table 6, which displays the performance of both the RMBFFT architecture and previous efforts, reveals that the proposed scheme provides higher performance in terms of power dissipation, energy consumption, throughput, reconfiguration time, and number of FPGA slices. In the proposed method, wl and P represent word-length and the number of processing units (MPRP) of RMBFFT architecture; both can be chosen in accordance with the required throughput. For example, using four processing units ($P=4$), the achieved sampling rate is large enough to handle the maximum required throughput of LTE (30.72 MS/s) [44] when the word-length equals 16 ($wl=16$). ASIC implementation of this architecture using CMOS technology can increase processing speed and consequently decrease the number of processing units (P) required. The energy consumption per symbol for the proposed RMBFFT with size 512 ($N=512$) is estimated to be 7, 15, 26, and 40 pJ when the word-length (wl) equals 8, 12, 16, and 20 bits respectively.

Reducing FFT word-length decreases energy consumption but causes a loss of accuracy or SQNR. Therefore, OFDM systems need FFT modules to be implemented with shorter word-lengths while meeting the required SQNR [9]. The SQNR performance of the proposed RMBFFT with $N=512$ is calculated and compared to those of the conventional FFT processors and shown in Table 7. The proposed scheme can be optimized based on the requirements at run-time, while the hardware can be reconfigured with a proper word-length and reasonable SQNR to guarantee the minimum required BER. For exam-

Table 7 SQNR performance comparison with previous works

FFT Design	[22]	[25]	[53]	[11]	[23]	Proposed RMBFFT			
Architecture	SDF	pipeline	SDF	MDF	memory-based	memory-based			
FFT size	8192	8192	1024	512	512	512	512	512	512
Word-length (bits)	16	16	16	14	12	8	12	16	20
SQNR (dB)	34.9	53	51	41	57	16	41	61	69

ple, when the communication system works with a small FFT size ($N=64$) and BPSK modulation in a channel with high SNR, 8-bit word-length is quite enough to provide the required SQNR and BER. On the other hand, when higher SQNR is needed, longer word-lengths can be dynamically chosen in RMBFFT.

6 Conclusion

This paper proposes RMBFFT, a multi-precision reconfigurable FFT architecture, as a dynamic trade-off between accuracy and energy efficiency in OFDM systems. RMBFFT is a memory-based DIF radix-4 FFT that can be reconfigured according to the minimum required word-length based on the system specifications, including BER, SNR, the modulation scheme, and FFT size at run-time. Quantization noise analysis is exploited for RMBFFT and the required word-lengths are calculated statistically. To validate the analysis, the proposed RMBFFT was simulated and applied to an OFDM transceiver in AWGN and Rayleigh fading channels. The results of the simulation followed a similar trend as seen in the analytic calculations. Synthesizing the basic operators of RMBFFT on an FPGA showed more than an 80% energy saving with RMBFFT over the traditional implementation. In addition, introducing a reconfigurable memory organization, can reduce the energy consumption of RMBFFT memory by approximately 60%.

In the future, we plan to implement the proposed RMBFFT in an OFDM system and measure the energy savings precisely. RMBFFT can be applied in other applications such as image processing to increase processing speed and energy efficiency. This reconfigurable architecture can be used in other error-tolerable applications.

References

1. R. Airoldi, F. Campi, M. Cucchi, D. Revanna, O. Anjum, J. Nurmi, Design and implementation of a power-aware FFT core for OFDM-based DSA-enabled cognitive radios. *J. Signal Process. Syst.* **78**(3), 257–265 (2015). DOI 10.1007/s11265-014-0894-z
2. A.S. Beulet Paul, S. Raju, R. Janakiraman, Low power reconfigurable FP-FFT core with an array of folded DA butterflies. *EURASIP J. Adv. Signal Process* **2014**(1), 1–17 (2014). DOI 10.1186/1687-6180-2014-144.
3. S.S. Bhattacharyya, E.F. Deprettere, R. Leupers, J. Takala, *Optimization of number representation. Handbook of Signal Processing Systems*. Springer, New York (2010)

4. H. Bogucka, P. Kryszkiewicz, A. Kliks, Dynamic spectrum aggregation for future 5G communications. *IEEE Commun. Mag.* **53**(5), 35–43 (2015). DOI 10.1109/MCOM.2015.7105639
5. G. Caffarena, C. Carreras, J.A. López, Á. Fernández, SQNR estimation of fixed-point DSP algorithms. *EURASIP J. Adv. Signal Process* **2010**, 21:1–21:12 (2010). DOI 10.1155/2010/171027.
6. C.H. Chang, C.L. Wang, Y.T. Chang, A novel memory-based FFT processor for DMT/OFDM applications, in *IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 4, pp. 1921–1924 (1999). DOI 10.1109/ICASSP.1999.758300
7. W.H. Chang, T.Q. Nguyen, On the fixed-point accuracy analysis of FFT algorithms. *IEEE Trans. Signal Process.* **56**(10), 4673–4682 (2008). DOI 10.1109/TSP.2008.924637.
8. K.H. Chen, A low-memory-access length-adaptive architecture for 2^n -point FFT. *Circuits Syst Signal Process* **34**(2), 459–482 (2015).
9. Y. Chen, Y.C. Tsao, Y.W. Lin, C.H. Lin, C.Y. Lee, An indexed-scaling pipelined FFT processor for OFDM-Based WPAN applications. *IEEE Trans. Circuits Syst. II Exp. Briefs* **55**(2), 146–150 (2008).
10. I. Cho, T. Patyk, D. Guevorkian, J. Takala, S. Bhattacharyya, Pipelined FFT for wireless communications supporting 128-2048 / 1536-point transforms, in *Global Conference on Signal and Information Processing (IEEE GlobalSIP)*, pp. 1242-1245 (2013) DOI 10.1109/GlobalSIP.2013.6737133
11. T. Cho, H. Lee, J. Park, C. Park, A high-speed low-complexity modified radix-2⁵ FFT processor for gigabit WPAN applications, in *IEEE International Symposium of Circuits and Systems (ISCAS)* pp. 1259–1262 (2011).
12. F. Cladera, M. Gautier, O. Sentieys, Energy-aware computing via adaptive precision under performance constraints in OFDM wireless receivers, in *IEEE Computer Society Annual Symposium on VLSI*, pp. 591–596 (2015). DOI 10.1109/ISVLSI.2015.88
13. F. de Dinechin, B. Pasca, Designing custom arithmetic datapaths with FloPoCo. *IEEE Design and Test of Computers* **28**(4), 18–27 (2011). DOI 10.1109/MDT.2011.44
14. R. Duan, M. Bi, C. Gniady, Exploring memory energy optimizations in smartphones, in *Green Computing Conference and Workshops (IGCC)*, pp. 1–8 (2011)
15. D. Feng, C. Jiang, G. Lim, L.J. Cimini, G. Feng, G.Y. Li, A survey of energy-efficient wireless communications. *IEEE Commun. Surv. Tuts.* **15**(1), 167–178 (2013)
16. M.L. Ferreira, J.C. Ferreira, Reconfigurable NC-OFDM processor for 5G communications, in *IEEE 13th International Conference on Embedded and Ubiquitous Computing*, pp. 199–204 (2015). DOI 10.1109/EUC.2015.29.
17. M.L. Ferreira, A. Barahimi, J.C. Ferreira, Reconfigurable FPGA-Based FFT processor for cognitive radio applications, in *International Symposium on Applied Reconfigurable Computing*, Springer pp. 223–232 (2016).
18. M.P. Fitz, J.P. Seymour, On the bit error probability of QAM modulation. *Int. J. Wireless Inform. Networks* **1**(2), 131–139 (1994)
19. Y. Gijung, J. Yunho, Scalable FFT processor for MIMO-OFDM based SDR systems, in *5th IEEE International Symposium on Wireless Pervasive Computing (ISWPC)*, pp. 517–521 (2010)
20. X. Guan, Y. Fei, H. Lin, Hierarchical design of an application-specific instruction set processor for high-throughput and scalable FFT processing. *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.* **20**(3), 551–563 (2012)
21. S. Haykin, Cognitive radio: Brain-empowered wireless communications. *IEEE J. Sel. Areas Commun.* **23**(2), 201–220 (2005). DOI 10.1109/JSAC.2004.839380
22. J. He, J. Wang, X. Xu, Word-length optimization of a pipelined FFT processor, in *IEEE International Conference on Consumer Electronics, Communications and Networks (CECNet)*, pp. 5485–5488 (2011).
23. S.J. Huang, S.G. Chen, A high-throughput radix-16 FFT processor with parallel and normal input/output ordering for IEEE 802.15.3c systems. *IEEE Trans. Circuits Syst. I Reg. Papers* **59**(8), 1752–1765 (2012).
24. N. Janakiraman, P. Nirmalkumar, S. M. Akram, Coarse grained ADRES based MIMO-OFDM transceiver with new radix-2⁵ pipeline FFT/IFFT processor. *Circuits Syst Signal Process* **34**(3), 851–873 (2015).

25. S. Johansson, S. He, P. Nilsson, Wordlength optimization of a pipelined FFT processor, in *Proceedings of the 42nd Midwest Symposium on Circuits and Systems*, pp. 501–503 (1999).
26. J. Kim, S. Yoshizawa, Y. Miyana, Dynamic wordlength calibration to reduce power dissipation in wireless OFDM systems, in *IEEE Asia Pacific Conference on Circuits and Systems (APCCAS)*, pp. 628–631 (2010). DOI 10.1109/APCCAS.2010.5774944.
27. J. Kim, S. Yoshizawa, Dynamic wordlength calibration for energy reduction FFT processors in wireless LAN, in *IEEE International Midwest Symposium on Circuits and Systems (MWSCAS)*, pp. 1–4 (2011).
28. P. Korkmaz, B.E.S. Akgul, K.V. Palem, Energy , performance , and probability tradeoffs for energy-efficient probabilistic CMOS circuits. *IEEE Trans. Circuits Syst. I Reg. Papers* **55**(8), 2249–2262 (2008).
29. S. Lee, A. Gerstlauer, Fine grain precision scaling for datapath approximations in digital signal processing systems, in *FIP/IEEE International Conference on Very Large Scale Integration-System on a Chip*, pp. 119–143. Springer (2015). DOI 10.1007/978-3-319-23799-2
30. T.Y. Lee, C.H. Huang, W.C. Chen, M.J. Liu, A low-area dynamic reconfigurable MDC FFT processor design. *Microprocess. Microsyst.* **42**, 227–234 (2016).
31. D. Menard, O. Sentieys, DSP code generation with optimized data word-length selection, in *International Workshop on Software and Compilers for Embedded Systems*, pp. 214–228. Springer (2004).
32. R. Nehmeh, D. Menard, E. Nogues, A. Banciu, T. Michel, R. Rocher, Fast integer word-length optimization for fixed-point systems. *J. Signal Process. Syst.*, pp. 1–16 (2015). DOI 10.1007/s11265-015-0990-8
33. H.N. Nguyen, D. Menard, O. Sentieys, Dynamic precision scaling for low power WCDMA receiver, in *IEEE International Symposium on Circuits and Systems*, pp. 205–208 (2009).
34. D. Novo, B. Bougard, A. Lambrechts, L.V.D. Perre, F. Catthoor, L. der Perre, F. Catthoor, Scenario-based fixed-point data format refinement to enable energy-scalable software defined radios, in *Proceedings of the Conference on Design, Automation and Test in Europe, DATE'08*, pp. 722–727. ACM, New York, NY, USA (2008). DOI 10.1145/1403375.1403550.
35. A.V. Oppenheim, R.W. Schaffer, *Discrete-Time Signal Processing*, 3rd ed. Prentice-Hall (2009).
36. B. Parhami, *Computer Arithmetic: Algorithms and Hardware Designs*, 2nd ed. Oxford University Press, New York (2010).
37. T. Patyk, D. Guevorkian, T. Pitkanen, P. Jaaskelainen, J. Takala, Low-power application-specific FFT processor for LTE applications, in *International Conference on Embedded Computer Systems: Architectures, Modeling, and Simulation (SAMOS)*, pp. 28–32 (2013). DOI 10.1109/SAMOS.2013.6621102.
38. A. Pedram, J.D. McCalpin, A. Gerstlauer, A highly efficient multicore floating-point FFT architecture based on hybrid linear algebra/FFT cores. *J. Signal Process. Syst.* **77**(1-2), 169–190 (2014). DOI 10.1007/s11265-014-0896-x.
39. J.G. Proakis, *Digital Signal Processing: Principles, Algorithms, and Applications*, 3rd ed. Prentice-Hall (2000).
40. J.G. Proakis, M. Salehi, *Digital Communications*, 5th ed. McGraw Hill (2007).
41. R. Rajbanshi, A.M. Wyglinski, G.J. Minden, An efficient implementation of NC-OFDM transceivers for cognitive radios, in *1st International Conference on Cognitive Radio Oriented Wireless Networks and Communications*, pp. 1–5 (2006). DOI 10.1109/CROWNCOM.2006.363452
42. R. Rocher, D. Menard, P. Scalart, O. Sentieys, Analytical approach for numerical accuracy estimation of fixed-point systems based on smooth operations. *IEEE Trans. Circuits Syst. I Reg. Papers* **59**(10), 2326–2339 (2012).
43. A. Sampson, W. Dietl, E. Fortuna, D. Gnanapragasam, L. Ceze, D. Grossman, EnerJ: approximate data types for safe and general low-power computation, in *Proceedings of the 32nd ACM SIGPLAN Conference on Programming Language Design and Implementation, PLDI '11*, pp. 164–174. ACM, New York, NY, USA (2011). DOI 10.1145/1993498.1993518.

44. V. Sarada, T. Vigneswaran, Reconfigurable FFT processor - A broader perspective survey. *Int. J. Eng. Technol.* **5**(2), 949–956 (2013).
45. M.K. Simon, M.S. Alouini, *Digital Communication Over Fading Channels*. John Wiley & Sons (2005)
46. S. Stotas, A. Nallanathan, On the throughput and spectrum sensing enhancement of opportunistic spectrum access cognitive radio networks. *IEEE Trans. Wireless Commun.* **11**(1), 97–107 (2012)
47. C.H. Van Berkel, Multi-core for mobile phones, in *Proceedings of the Conference on Design, Automation and Test in Europe*, pp. 1260–1265 (2009)
48. C. Vennila, G. Lakshminarayanan, S.B. Ko, Dynamic partial reconfigurable FFT for OFDM based communication systems. *Circuits Syst Signal Process* **31**(3), 1049–1066 (2012). DOI 10.1007/s00034-011-9367-9
49. L. Wilhelmsson, I. Diaz, T. Olsson, V. Owall, Analysis of a novel low complex SNR estimation technique for OFDM systems, in *IEEE Wireless Communications and Networking Conference (WCNC)*, pp. 1646–1651 (2011)
50. M. Woh, S. Mahlke, T. Mudge, Mobile supercomputers for the next generation cell phone. *Computer* **43**(1), 81–85 (2010)
51. H. Xiao, A. Pan, Y. Chen, X. Zeng, Low-cost reconfigurable VLSI architecture for fast Fourier transform. *IEEE Trans. Consum. Electron* **54**(4), 1617–1622 (2008). DOI 10.1109/TCE.2008.4711210
52. Xilinx Inc: XPower Analyzer (2011). URL http://www.xilinx.com/products/design_tools/logic_design/verification/xpower.htm
53. C. Yang, Y.Z. Xie, L. Chen, H. Chen, Y. Deng, Design of a configurable fixed-point FFT processor, in *IET International Radar Conference*, pp. 1–4 (2015).
54. S. Yoshizawa, Y. Miyanaga, Use of a variable wordlength technique in an OFDM receiver to reduce energy dissipation. *IEEE Trans. Circuits Syst. I Reg. Papers* **55**(9), 2848–2859 (2008). DOI 10.1109/TCSI.2008.920098.